

L0.1 Why Learn Assembly, L0.2 Computer and Assembly Language Quiz

1. Which statement best captures why assembly isn't "just another language"?
 - A. It is easier to read than C
 - B. It is an interface between hardware and software and underlies HLL implementations
 - C. It is fully portable across architectures
 - D. It eliminates bugs due to its low-level nature

ANS:
2. Which of the following is a common disadvantage of writing large applications entirely in assembly?
 - A. Excessive compiler optimization
 - B. Difficult to develop, read, and maintain; not portable
 - C. Inability to access special instructions
 - D. Incompatibility with embedded systems

ANS:
3. Which scenario typically requires assembly on ARM Cortex-M?
 - A. Generic arithmetic in a desktop application
 - B. Device driver sections requiring precise register/I/O access
 - C. High-level GUI layout code
 - D. Standard library string functions

ANS:
4. What is the primary reason startup code for microcontrollers is often in assembly?
 - A. HLLs cannot declare global variables
 - B. It must set stack/heap, define the vector table, and default ISRs at reset
 - C. C compilers cannot initialize constants
 - D. Assembly is required for loops

ANS:
5. Which example shows how understanding data representation avoids logic errors when comparing signed and unsigned values?
 - A. Using volatile on all variables
 - B. Casting unsigned uint32_t x to signed int32_t before comparing with a signed int32_t y
 - C. Always using uint32_t for all variables
 - D. Turning on compiler optimizations

ANS:
6. Which registers are general-purpose on ARM Cortex-M?
 - A. R0–R12
 - B. R13–R15
 - C. CONTROL, BASEPRI, PRIMASK
 - D. xPSR only

ANS:
7. What are the roles of R13, R14, and R15 on ARM Cortex-M?
 - A. General purpose
 - B. Program counter, stack pointer, and link register in that order
 - C. Stack pointer (R13), link register (R14), program counter (R15)
 - D. Interrupt mask registers

ANS:

8. What does the instruction BX lr do in a typical function epilogue on Cortex-M?

- A. Saves lr to the stack
- B. Branches to the address in lr and sets execution state
- C. Loads the next instruction into lr
- D. Disables interrupts

ANS:

9. Which statement about instruction fetch is accurate for Cortex-M?

- A. The PC increments by 2 for every 16-bit instruction fetch
- B. The PC always increments by 4 because instruction fetches bring 4 bytes (either two 16-bit or one 32-bit instruction)
- C. The PC increments by 8 on 32-bit instructions
- D. The PC increments by 1 byte on unaligned code

ANS:

10. In a Harvard architecture microcontroller like Cortex-M, which is a main advantage compared to pure von Neumann for DSP-like workloads?

- A. Single shared memory reduces complexity
- B. Simultaneous instruction and data fetch increase bandwidth and predictability
- C. Unified address space aids code density
- D. Automatic cache coherence

ANS:

11. What is the size of each memory location in a byte-addressable memory system as described?

- A. 1 bit
- B. 8 bits
- C. 16 bits
- D. 32 bits

ANS: